

*In the Claims*

The status of claims in the case is as follows:

1       1. [Previously presented] A method for control and  
2       management of communication traffic, comprising the steps  
3       of:

4               expressing access rules as filters referencing system  
5               kernel data;

6               for outbound processing, determining source application  
7               indicia;

8               for inbound packet processing, executing a look-ahead  
9               function to determine target application indicia; said  
10          look-ahead function being executed within a protocol  
11          stack including an IP layer, a transport layer, a  
12          sockets layer, and an application layer and which, for  
13          said inbound packet, said IP layer provides to said  
14          transport layer said inbound packet, marked as non-  
15          deliverable, and receives back from said transport  
16          layer indicia, provided to said transport layer by said

17       sockets layer, identifying the application layer  
18       application to which said packet would have been  
19       delivered; and

20           responsive to said source or target application  
21       indicia, executing filter processing; said filter  
22       processing including constructing and evaluating  
23       logical expressions of arbitrary length, and  
24       selectively using a set of logical operators,  
25       alternative filter selector fields, and value set.

1       2. [Currently amended] The method of claim 1, wherein  
2       said protocol stack is a TCP/IP protocol stack, and further  
3       comprising the steps of executing said determining and  
4       executing steps within a kernel filtering function upon  
5       encountering a filter selector field referencing kernel data  
6       not included in said packet.

1       3. [Currently amended] The method of claim 1, wherein  
2       said protocol stack is a TCP/IP protocol stack, and said  
3       filter processing including the steps of:

4           determining a task or thread identifier;

5           based on said task or thread identifier, determining a  
6           process or job identifier; and

7           based on said process or job identifier, determining  
8           job or process attributes for filter processing.

1       4. [Currently amended] The method of claim 1, wherein  
2       said protocol stack is a TCP/IP protocol stack, and said  
3       filter processing including the steps of:

4           determining a user identifier; and

5           based on said user identifier, determining user  
6           attributes for filter processing.

1       5. [Original] The method of claim 3, further comprising  
2       the step of determining from said task identifier a work  
3       control block containing said process or job identifier.

1       6. [Canceled]

2       7. [Canceled]

1       8. [Currently amended] The method of claim 1, wherein  
2       said protocol stack is a TCP/IP protocol stack, and further  
3       comprising the steps of:

4           delivering to said filters infrastructure access rules  
5           for defining security context.

1       9. [Original] The method of claim 8, said infrastructure  
2       including logging, auditing, and filter rule load controls.

1       10. [Previously presented] A method for control and  
2       management of aspects of communication traffic within  
3       filtering, comprising the steps of:

4           receiving IP packet data into a TCP/IP protocol stack  
5           executing within a system kernel;

6           for an inbound IP packet, executing a look-ahead  
7           function within a protocol stack including an IP layer,  
8           a transport layer, a sockets layer, and an application  
9           layer and which, for said IP inbound packet, said IP  
10          layer provides to said transport layer said inbound IP  
11          packet, marked as non-deliverable, and receives back  
12          from said transport layer indicia, provided to said

13           transport layer by said sockets layer, identifying the  
14           application layer application to which said packet  
15           would have been delivered; and

16           executing filtering code within said system kernel with  
17           respect to non-IP packet data accessed within said  
18           system kernel outside of said TCP/IP protocol stack;  
19           said filtering code constructing and evaluating logical  
20           expressions of arbitrary length, and selectively using  
21           a set of logical operators, alternative filter selector  
22           fields, and value set.

1       11. [Original] The method of claim 10, said non-IP packet  
2       data including context data regarding said IP packet.

1       12. [Original] The method of claim 10, said non-IP packet  
2       data including data specific to a task generating said non-  
3       IP packet data.

1       13. [Original] The method of claim 10, said non-IP packet  
2       data including data specific to a task that will receive  
3       said IP packet.

1       14. [Original] The method of claim 11, said context data

2 including packet arrival interface indicia.

1 15. [Canceled]

2 16. [Canceled]

3 17. [Original]

4 18. [Previously presented] A method for centralizing  
5 system-wide communication management and control within  
6 filter rules, comprising the steps of:

7 providing filter statements syntax for accepting  
8 parameters in the form of a selector, each selector  
9 specifying selector field, operator, and a set of  
10 values;

11 for an inbound packet, executing a look-ahead function  
12 within a protocol stack including an IP layer, a  
13 transport layer, a sockets layer, and an application  
14 layer and which, for said inbound packet, said IP layer  
15 provides to said transport layer said inbound packet,  
16 marked as non-deliverable, and receives back from said  
17 transport layer indicia, provided to said transport  
18 layer by said sockets layer, identifying the  
19 application layer application to which said packet

20           would have been delivered by said sockets layer;

21           said selector referencing data that does not exist in

22           IP packets;

23           processing said filter statements, including

24           constructing and evaluating logical expressions of

25           arbitrary length, and selectively using a set of

26           logical operators, alternative filter selector fields,

27           and value set.

1       19. [Currently amended] The method of claim 18, wherein

2       said protocol stack is a TCP/IP protocol stack, and said

3       parameters selectively including userid, user profile, user

4       class, user group, user group authority, user special

5       authority, job name, process name, job group, job class, job

6       priority, other job or process attributes, and date & time.

1       20. [Currently amended] The method of claim 18, wherein

2       said protocol stack is a TCP/IP protocol stack, and said

3       filters statements being provided within a user interface to

4       said system.

1       21. [Currently amended] The method of claim 18, wherein

2       said protocol stack is a TCP/IP protocol stack, and further  
3       comprising the steps of:

4             establishing a tunnel between two IP address limiting  
5             traffic to applications bound to ports at each end of  
6             said tunnel;

7             said filtering code accessing filtering attributes  
8             further limiting traffic selectively to job indicia;  
9             and

10           operating said filtering code within a kernel filtering  
11           function upon encountering a filter selector field  
12           referencing kernel data not included in said traffic.

1       22. [Currently amended] A method for traversing a portion  
2       only of a protocol stack to disallow selective IP packet  
3       traffic, comprising the steps of:

4             receiving a packet in the kernel of the operating  
5             system of a first node from an application, said kernel  
6             including a filter processor; said filter processor for  
7             constructing and evaluating logical expressions of  
8             arbitrary length, said logical expressions selectively

9           including a set of logical operators, alternative  
10          filter selector fields, and value set;

11         for inbound packet processing to a first node from a  
12          second node, executing a look-ahead function in the  
13          system kernel of said first node to determine a target  
14          application; said system kernel including a TCP/IP  
15          protocol stack including an IP layer, a transport  
16          layer, a sockets layer, and an application layer and  
17          which, for said inbound packet, said IP layer provides  
18          to said transport layer said inbound packet, marked as  
19          non-deliverable, and receives back from said transport  
20          layer indicia identifying the application layer  
21          application to which said packet would have been  
22          delivered;

23         for both said inbound packet processing, and for  
24          outbound packet processing from said first node to said  
25          second node, executing within said kernel the steps of

26           processing said packet by determining a task ID;

27           responsive to said task ID, determining a  
28          corresponding work control block;

29                   determining a user ID, process or job identifier  
30                   from said work control block;  
  
31                   from the user ID, process or job identifier  
32                   selectively determining attributes for said user  
33                   process or job; and  
  
34                   passing said attributes to said filter processor  
35                   for managing and controlling communication  
36                   traffic.

1       23. [Previously presented] A method for expressing access  
2       rules as filters, comprising the steps of:  
  
3                   providing a filter statements syntax for accepting  
4                   parameters in the form of a selector, each selector  
5                   specifying selector field, operator, and a set of  
6                   values; and  
  
7                   said selector referencing data that does not exist in  
8                   IP packets for controlling access to an application;  
  
9                   for an inbound IP packet, executing a look-ahead  
10                  function within a protocol stack including an IP layer,

11           a transport layer, a sockets layer, and an application  
12           layer and which, for said IP inbound packet, said IP  
13           layer provides to said transport layer said inbound IP  
14           packet, marked as non-deliverable, and receives back  
15           from said transport layer indicia, provided to said  
16           transport layer by said sockets layer, identifying the  
17           application layer application to which said packet  
18           would have been delivered; and

19           processing said filter statements by constructing and  
20           evaluating logical expressions of arbitrary length,  
21           said logical expressions selectively including a set of  
22           logical operators, alternative filter selector fields,  
23           and value set referencing said application layer  
24           application.

1       24. [Previously presented] A method for managing and  
2       controlling communication traffic by centralizing access  
3       rules in filters executing within and referencing data  
4       available in system kernels, comprising the steps for  
5       outbound packet processing from a first node to a second  
6       node of:

7           receiving said packet in the kernel of the operating

8           system of said first node from an application or  
9       process at said first node;  
  
10           processing said packet by determining a task ID;  
  
11           responsive to said task ID, determining a corresponding  
12       work control block;  
  
13           responsive to said work control block, determining a  
14       process or job identifier;  
  
15           responsive to said process or job identifier,  
16       determining job or process attributes; and  
  
17           executing said filters by constructing and evaluating  
18       logical expressions of arbitrary length, said logical  
19       expressions selectively including a set of logical  
20       operators, alternative filter selector fields, and  
21       value set.

1       25. [Previously presented] The method of claim 24, further  
2       comprising the steps for inbound packet processing from said  
3       second node to said first node of:

4           initially operating said kernel at said first node to  
5           determine a target application for said packet at said  
6           first node by executing a look-ahead function within a  
7           protocol stack including an IP layer, a transport  
8           layer, a sockets layer, and an application layer and  
9           which, for said inbound packet, said IP layer provides  
10          to said transport layer said inbound packet, marked as  
11          non-deliverable, and receives back from said transport  
12          layer indicia, provided to said transport layer by said  
13          sockets layer, identifying the application layer  
14          application to which said packet would have been  
15          delivered;.

26. [Canceled]

1 27. [Canceled]

2 28. [Canceled]

1 29. [Currently amended] A method for managing and  
2 controlling communication traffic by centralizing the access  
3 rules, comprising the steps for outbound packet processing  
4 from a first node to a second node of:

5 receiving said packet in the kernel of the operating  
6 system of said first node from an application or

7           process at said first node, said kernel including a  
8           filter processor for constructing and evaluating  
9           logical expressions of arbitrary length, said logical  
10          expressions selectively including a set of logical  
11          operators, alternative filter selector fields, and  
12          value set;

13         processing said packet within a TCP/IP stack;  
14           by determining a task ID;  
15           responsive to said task ID, determining a  
16           corresponding work control block;  
17           determining a user ID control block from said work  
18           control block;  
19           from the user ID control block determining  
20           attributes for said user; and  
21           passing said attributes to said filter processor  
22           for managing and controlling communication  
23           traffic.

1       30. [Currently amended] The method of claim 29, further  
2 comprising the steps for inbound packet processing from said  
3 second node to said first node of:

4             initially operating said kernel at said first node to  
5 determine a target application for said packet at said  
6 first node by executing a look-ahead function within a  
7 protocol said TCP/IP protocol stack including an IP  
8 layer, a transport layer, a sockets layer, and an  
9 application layer and which, for said inbound packet,  
10 said IP layer provides to said transport layer said  
11 inbound packet, marked as non-deliverable, and receives  
12 back from said transport layer indicia, provided to  
13 said transport layer by said sockets layer, identifying  
14 the application layer application to which said packet  
15 would have been delivered.

1       31. [Canceled]  
2       32. [Canceled]  
3       33. [Canceled]

1       34. [Previously presented] A method for control and  
2 management of communication traffic with respect to a system  
3 node, comprising the steps of:

4 receiving at said system node an inbound packet; and

5 executing within a protocol stack of the system kernel

6 of said system node a filtering function identifying

7 for said inbound packet a filter referencing non-packet

8 data, and constructing and evaluating logical

9 expressions of arbitrary length, said logical

10 expressions selectively including a set of logical

11 operators, alternative filter selector fields, and

12 value set; and

13 responsive to said filter, executing a look-ahead

14 function for identifying a target application for said

15 inbound packet; said look-ahead function executed

16 within a protocol stack including an IP layer, a

17 transport layer, a sockets layer, and an application

18 layer and which, for said IP inbound packet, said IP

19 layer provides to said transport layer said inbound

20 packet, marked as non-deliverable, and receives back

21 from said transport layer indicia, provided to said

22 transport layer by said sockets layer, identifying the

23 application layer application to which said packet

24 would have been delivered;.

1       35. [Currently amended] The look-ahead function of the  
2       method of claim 34 wherein said protocol stack is a TCP/IP  
3       protocol stack, and further comprising the steps of:

4              passing to a transport layer function identified by an  
5              IP header a packet marked non-deliverable for  
6              determining which user-level process or job is to  
7              receive said packet;

8              receiving from said transport layer an application  
9              layer task identifier for said user-level process or  
10             job; and thereafter

11             passing said packet marked by said task identifier to  
12             said transport layer for delivery to said application  
13             layer task.

1       36. [Currently amended] System for control and management  
2       of communication traffic, comprising:

3              a system kernel including a filter function and stack  
4              data;

5              said filter function including a filter selectively

6 referencing said stack data for expressing access  
7 rules;

8 said filter function being responsive to receipt of an  
9 outbound packet for determining a source application;

10 said filter function being responsive to receipt of an  
11 inbound packet processing for executing a look-ahead  
12 function within a TCP/IP protocol stack to determine a  
13 target application; said protocol stack including an IP  
14 layer, a transport layer, a sockets layer, and an  
15 application layer and which, for said inbound packet,  
16 said IP layer provides to said transport layer said  
17 inbound packet, marked as non-deliverable, and receives  
18 back from said transport layer indicia, provided to  
19 said transport layer by said sockets layer, identifying  
20 the application layer application to which said packet  
21 would have been delivered; and

22 said filter function being responsive to said source or  
23 target application for executing filter processing  
24 including constructing and evaluating logical  
25 expressions of arbitrary length, said logical  
26 expressions selectively including a set of logical

27           operators, alternative filter selector fields, and  
28           value set.

1       37. [Previously presented] A system for control and  
2       management of aspects of communication traffic within  
3       filtering, comprising:

4           a system kernel;

5           a protocol stack including an IP layer, a transport  
6       layer, a sockets layer, and an application layer for  
7       executing within said system kernel, responsive to an  
8       inbound IP packet, a look-ahead function by which said  
9       IP layer provides to said transport layer said inbound  
10      IP packet, marked as non-deliverable, and receives back  
11      from said transport layer indicia, provided to said  
12      transport layer by said sockets layer, identifying the  
13      application layer application to which said packet  
14      would have been delivered; and

15       filtering code within said system kernel operable with  
16       respect to non-IP packet data accessed within said  
17       system kernel outside of said protocol stack for  
18       controlling and managing said aspects of communication

19           traffic; said filter code for constructing and  
20           evaluating logical expressions of arbitrary length,  
21           said logical expressions selectively including a set of  
22           logical operators, alternative filter selector fields,  
23           and value set.

1       38. [Previously presented] A system for centralizing  
2           system-wide communication management and control within  
3           filter rules, comprising:

4           filter statements having a syntax for accepting  
5           parameters in the form of a selector, each selector  
6           specifying selector field, operator, and a set of  
7           values;

8           said selector referencing data that does not exist in  
9           IP packets;

10           a look-ahead function within a protocol stack including  
11           an IP layer, a transport layer, a sockets layer, and an  
12           application layer which, for an inbound packet, said IP  
13           layer provides to said transport layer said inbound  
14           packet, marked as non-deliverable, and receives back  
15           from said transport layer indicia, provided to said

16           transport layer by said sockets layer, for identifying  
17           the application layer application to which said packet  
18           would have been delivered; and

19           a filter processor for constructing and evaluating  
20           filter statements including logical expressions of  
21           arbitrary length, said logical expressions selectively  
22           including a set of logical operators, alternative  
23           filter selector fields, and value set.

1       39. [Currently amended] A system for traversing a portion  
2       only of a TCP/IP protocol stack to disallow selective IP  
3       packet traffic, comprising:

4           a system kernel;

5           a filter processor executing within said system kernel  
6           for constructing and evaluating logical expressions of  
7           arbitrary length, said logical expressions selectively  
8           including a set of logical operators, alternative  
9           filter selector fields, and value set;

10          said filter processor responsive to an inbound packet  
11          for executing a look-ahead function for determining a

12           target application; said look-ahead function operating  
13        within [[a]] said TCP/IP protocol stack including an IP  
14        layer, a transport layer, a sockets layer, and an  
15        application layer and which, for said IP inbound  
16        packet, said IP layer provides to said transport layer  
17        said inbound IP packet, marked as non-deliverable, and  
18        receives back from said transport layer indicia,  
19        provided to said transport layer by said sockets layer,  
20        identifying the application layer application to which  
21        said packet would have been delivered;

22           said filter processor responsive to both inbound and  
23        outbound packets for

24           processing said packet by determining a task ID;

25           responsive to said task ID, determining a  
26        corresponding work control block;

27           determining a user ID, process or job identifier  
28        from said work control block;

29           from the user ID, process or job identifier  
30        selectively determining attributes for said user

31                   process or job; and  
  
32                   passing said attributes to said filter processor  
33                   for managing and controlling communication  
34                   traffic.

1       40. [Previously presented] A system for expressing access  
2       rules as filters, comprising:

3                   filter statements for accepting parameters in the form  
4                   of a selector, each selector specifying selector field,  
5                   operator, and a set of values;

6                   said selector referencing data that does not exist in  
7                   IP packets for controlling access to an application;

8                   a look-ahead function executing within a protocol stack  
9                   including an IP layer, a transport layer, a sockets  
10                  layer, and an application layer and which, for an  
11                  inbound packet, said IP layer provides to said  
12                  transport layer said inbound packet, marked as non-  
13                  deliverable, and receives back from said transport  
14                  layer indicia, provided to said transport layer by said  
15                  sockets layer, identifying the application layer

16 application to which said packet would have been  
17 delivered; and

18 a filter processor for constructing and evaluating said  
19 filter statements as logical expressions of arbitrary  
20 length, each said logical expression selectively  
21 including said operator selected from a set of logical  
22 operators, alternative filter selector fields, and  
23 value set.

1 41. [Currently amended] A system for managing and  
2 controlling communication traffic by centralizing access  
3 rules in filters executing within and referencing data  
4 available in system kernels, comprising:

5 a computer readable medium;

6 first code for receiving a packet in the kernel of the  
7 operating system of a first node from an application or  
8 process at said first node; said kernel responsive to  
9 an inbound packet, for executing a look-ahead function  
10 within a TCP/IP protocol stack including an IP layer, a  
11 transport layer, a sockets layer, and an application  
12 layer and which, for said inbound packet, said IP layer

13 provides to said transport layer said inbound IP  
14 packet, marked as non-deliverable, and receives back  
15 from said transport layer indicia, provided to said  
16 transport layer by said sockets layer, identifying the  
17 application layer application to which said packet  
18 would have been delivered;

19 second code for processing said packet by determining a  
20 task ID;

21 third code responsive to said task ID for determining a  
22 corresponding work control block;

23 fourth code responsive to said work control block for  
24 determining a process or job identifier;

25 fifth code responsive to said process or job identifier  
26 for determining job or process attributes;

27 sixth code for executing said filters by constructing  
28 and evaluating logical expressions of arbitrary length,  
29 said logical expressions selectively including a set of  
30 logical operators, alternative filter selector fields,  
31 and value set; and wherein

32           said first, second, third, fourth, fifth, and sixth  
33        code is recorded on said computer readable medium.

1        42. [Canceled]

2        43. [Previously presented] A system for control and  
3        management of communication traffic with respect to a system  
4        node, comprising:

5           a filtering function executing within a protocol stack  
6        of the system kernel of said system node identifying  
7        for an inbound packet a filter referencing non-packet  
8        data; and

9           a look-ahead function responsive to said filter for  
10        identifying a target application for said inbound  
11        packet; said look-ahead function functioning within a  
12        protocol stack including an IP layer, a transport  
13        layer, a sockets layer, and an application layer and  
14        which, for said inbound packet, said IP layer provides  
15        to said transport layer said inbound packet, marked as  
16        non-deliverable, and receives back from said transport  
17        layer indicia, provided to said transport layer by said  
18        sockets layer, identifying the application layer

19 application to which said packet would have been  
20 delivered; ; and

21 a filter processor for constructing and evaluating  
22 logical expressions of arbitrary length, said logical  
23 expressions selectively including a set of logical  
24 operators, alternative filter selector fields, and  
25 value set.

44. [Canceled]

1 45. [Previously presented] A computer program product for  
2 control and management of aspects of communication traffic  
3 within filtering, said computer program product comprising:  
4 a computer readable medium;  
5 first program instructions to receive IP packet data  
6 into a TCP/IP protocol stack executing within a system  
7 kernel including, for processing an inbound IP packet,  
8 a look-ahead function within a protocol stack including  
9 an IP layer, a transport layer, a sockets layer, and an  
10 application layer and which, for said IP inbound  
11 packet, said IP layer provides to said transport layer

12           said inbound IP packet, marked as non-deliverable, and  
13           receives back from said transport layer indicia,  
14           provided to said transport layer by said sockets layer,  
15           identifying the application layer application to which  
16           said packet would have been delivered;

17           second program instructions to execute filtering code  
18           within said system kernel with respect to non-IP packet  
19           data accessed within said system kernel outside of said  
20           TCP/IP protocol stack by constructing and evaluating  
21           logical expressions of arbitrary length, said logical  
22           expressions selectively including a set of logical  
23           operators, alternative filter selector fields, and  
24           value set; and wherein

25           said first and second program instructions are recorded  
26           on said medium.

1       46. [Previously presented] A computer program product for  
2           centralizing system-wide communication management and  
3           control within filter rules, said computer program product  
4           comprising:

5           a computer readable medium;

6           first program instructions to execute filter statements  
7           having a syntax for accepting parameters in the form of  
8           a selector, each selector specifying selector field, a  
9           logical operator selected from a set of a plurality of  
10          logical operators, and a set of values; and

11         second program instructions to cause said selector to  
12         reference data that does not exist in IP packets, said  
13         data including application layer indicia obtained for  
14         an incoming packet by a look-ahead function; said look-  
15         ahead function executing within a protocol stack  
16         including an IP layer, a transport layer, a sockets  
17         layer, and an application layer and which, for said IP  
18         inbound packet, said IP layer provides to said  
19         transport layer said inbound IP packet, marked as non-  
20         deliverable, and receives back from said transport  
21         layer indicia, provided to said transport layer by said  
22         sockets layer, identifying the application layer  
23         application to which said packet would have been  
24         delivered; and wherein

25         said first and second program instructions are recorded  
26         on said medium.

1       47. [Previously presented] A computer program product for  
2 managing and controlling communication traffic by  
3 centralizing access rules in filters executing within and  
4 referencing data available in system kernels, said computer  
5 program product comprising:

6              a computer readable medium;

7              first program instructions to receive said packet in  
8              the kernel of the operating system of said first node  
9              from a process at said first node;

10             second program instructions to process said packet by  
11             determining a task ID;

12             third program instructions, responsive to said task ID,  
13             to determine a corresponding work control block;

14             fourth program instructions, responsive to said work  
15             control block, to determine a process or job  
16             identifier;

17             fifth program instructions, responsive to said process  
18             or job identifier, to determine job or process

19           attributes; and

20           sixth program instructions to execute a filter  
21           processor for constructing and evaluating logical  
22           expressions of arbitrary length, said logical  
23           expressions selectively including a set of logical  
24           operators, alternative filter selector fields, and  
25           value set; and wherein

26           said first, second, third, fourth, fifth, and sixth  
27           program instructions are recorded on said medium.

1       48. [Currently amended] The computer program product of  
2       claim 47, wherein said protocol stack is a TCP/IP protocol  
3       stack, and said computer program product further comprising  
4       for inbound packet processing from said second node to said  
5       first node:

6           sixth program instructions to initially operate said  
7           kernel at said first node to determine a target  
8           application for said packet at said first node by  
9           executing a look-ahead function within a protocol stack  
10          including an IP layer, a transport layer, a sockets  
11          layer, and an application layer and which, for said IP

12            inbound packet, said IP layer provides to said  
13            transport layer said inbound IP packet, marked as non-  
14            deliverable, and receives back from said transport  
15            layer indicia, provided to said transport layer by said  
16            sockets layer, identifying the application layer  
17            application to which said packet would have been  
18            delivered; ; and wherein

19            said sixth program instructions are recorded on said  
20            medium.

1        49. [Previously presented] A computer program product for  
2            control and management of communication traffic, comprising:  
3            a computer readable medium;  
4            first program instructions for expressing access rules  
5            as filters referencing system kernel data;  
6            second program instructions, for outbound processing,  
7            for determining a source application;  
8            third program instructions, for inbound packet  
9            processing, for executing a look-ahead function to

10 determine a target application; said look-ahead  
11 function operating within a protocol stack including an  
12 IP layer, a transport layer, a sockets layer, and an  
13 application layer and which, for said IP inbound  
14 packet, said IP layer provides to said transport layer  
15 said inbound IP packet, marked as non-deliverable, and  
16 receives back from said transport layer indicia,  
17 provided to said transport layer by said sockets layer,  
18 identifying the application layer application to which  
19 said packet would have been delivered;

20 fourth program instructions, selectively responsive to  
21 said source and target application, for executing  
22 filter processing including constructing and evaluating  
23 logical expressions of arbitrary length, said logical  
24 expressions selectively including a set of logical  
25 operators, alternative filter selector fields, and  
26 value set;; and wherein

27 said first, second, third, and fourth program  
28 instructions are recorded on said computer readable  
29 medium.

1 50. [Previously presented] A computer program product for

2 control and management of aspects of communication traffic  
3 within filtering, comprising:  
  
4 a computer readable medium;  
  
5 first program instructions for receiving IP packet data  
6 into a TCP/IP protocol stack executing within a system  
7 kernel;  
  
8 second program instructions for executing filtering  
9 code within said system kernel with respect to non-IP  
10 packet data accessed within said system kernel outside  
11 of said TCP/IP protocol stack; said filtering code  
12 constructing and evaluating logical expressions of  
13 arbitrary length, said logical expressions selectively  
14 including a set of logical operators, alternative  
15 filter selector fields, and value set; and wherein  
  
16 said first and second program instructions are recorded  
17 on said computer readable medium.

1 51. [Previously presented] A computer program element for  
2 centralizing system-wide communication management and  
3 control within filter rules, comprising:

4           a computer readable medium;

5           first program instructions for providing filter

6           statements syntax for accepting parameters in the form

7           of a selector, each selector specifying selector field,

8           a logical operator, and a set of values,

9           second program instructions for executing filtering by

10          constructing and evaluating logical expressions of

11          arbitrary length, said logical expressions selectively

12          including said logical operator selected from a set of

13          logical operators, at least one said selector field,

14          and at least one said value;

15          said selector referencing data that does not exist in

16          IP packets including data obtained, for an inbound IP

17          packet, by executing a look-ahead function within a

18          protocol stack including an IP layer, a transport

19          layer, a sockets layer, and an application layer and

20          which, for said IP inbound packet, said IP layer

21          provides to said transport layer said inbound IP

22          packet, marked as non-deliverable, and receives back

23          from said transport layer indicia, provided to said

24          transport layer by said sockets layer, identifying the

25 application layer application to which said packet  
26 would have been delivered; ; and wherein  
  
27 said first and second program instructions are recorded  
28 on said computer readable medium.

1 52. [Previously presented] A computer program product for  
2 managing and controlling communication traffic by  
3 centralizing access rules in filters executing within, and  
4 referencing data available in, system kernels, comprising:  
  
5 a computer readable medium;  
  
6 first program instructions for receiving said packet in  
7 the kernel of the operating system of said first node  
8 from an application or process at said first node;  
  
9 second program instructions for processing said packet  
10 by determining a task ID;  
  
11 third program instructions, responsive to said task ID,  
12 for determining a corresponding work control block;  
  
13 fourth program instructions, responsive to said work

14           control block, for determining a process or job  
15           identifier;  
  
16           fifth program instructions, responsive to said process  
17           or job identifier, for determining job or process  
18           attributes;  
  
19           sixth program instructions for executing a filter  
20           processor for constructing and evaluating logical  
21           expressions of arbitrary length, said logical  
22           expressions selectively including a set of logical  
23           operators, alternative filter selector fields, and  
24           value set; and wherein  
  
25           said first, second, third, fourth, fifth, and sixth  
26           program instructions are recorded on said computer  
27           readable medium.

1       53. [Previously presented] The computer program product of  
2       claim 52, further comprising for inbound packet processing  
3       from said second node to said first node:  
  
4           seventh program instructions initially operating said  
5           kernel at said first node to determine a target

6 application for said packet at said first node by  
7 executing a look-ahead function within a protocol stack  
8 including an IP layer, a transport layer, a sockets  
9 layer, and an application layer and which, for said IP  
10 inbound packet, said IP layer provides to said  
11 transport layer said inbound IP packet, marked as non-  
12 deliverable, and receives back from said transport  
13 layer indicia, provided to said transport layer by said  
14 sockets layer, identifying the application layer  
15 application to which said packet would have been  
16 delivered; and wherein  
  
17 said seventh program instructions are recorded on said  
18 computer readable medium.